DTIC FILE COPY

(4)

50272-101

| REPORT DOCUMENTATION PAGE | 1. REPORT NO. URTC-TP-01/87 | 2. | 3. Recipient's Accession No. |
|---|---|---|---|
| 4. Title and Subtitle Artificial Intelligence in the allocation of maintenance resources for intelligence systems | | | 5. Report Date Sept 87 publication |
| | | | 6. |
| 7. Author(s) Lester McConville, U.S. Army; Paul Orgren, Unisys Corp. | | | 8. Performing Organization Rept. No. |
| 9. Performing Organization Name and Address Unisys Corporation<br>Shipboard and Ground Systems<br>Technology Center<br>12010 Sunrise Valley Dr.<br>Reston, VA 22091 | | | 10. Project/Task/Work Unit No.<br>11. Contract(C) or Grant(G) No.<br>(C)<br>(G) |
| 12. Sponsori SAME | | | 13. Type of Report & Period Covered Final |
| AD-A197 557 | | | 14. |

15. Supplem

Presented at conference of Artificial Intelligence East 1987

16. Abstract (Limit: 200 words)

Commanders of Army military intelligence units in a tactical environment are responsible for keeping their assigned equipment maintained and in operation since early replacement in a conflict may not be possible. Scheduling maintenance assets to repair and service assigned equipment is critical to sucess. This paper discusses the development of an expert system to assist the commander in the proper allocation of his maintenance resources. Since this expert system assists in the resolution of an allocation of resources or scheduling problem, insight into the application of artificial intelligence to this type of problem is gained.



17. Document Analysis a. Descriptors

ARTIFICIAL INTELLIGENCE: MILITARY INTELLIGENCE: TACTICAL INTELLIGENCE: SCHEDULING: JOB SHOP SCHEDULING

b. Identifiers/Open-Ended Terms

EXPERT SYSTEM PLANNER: EXPERT SYSTEMS



c. COSATI Field/Group

| 18. Availability Statement | 19. Security Class (This Report) | 21. No. of Pages |
|---|---|---|
| Release unlimited | UNCLASSIFIED | 10 |
| | 20. Security Class (This Page) | 22. Price |
| | UNCLASSIFIED | 0.00 |

(ANSI-Z39.18) See Instructions on Reverse OPTIONAL FORM 272 (4-77) (Formerly NTIS-35) Department of Commerce

88 6 13 131

# DO NOT PRINT THESE INSTRUCTIONS AS A PAGE IN A REPORT

## INSTRUCTIONS

Optional Form 272, Report Documentation Page is based on Guidelines for Format and Production of Scientific and Technical Reports, ANSI Z39.18–1974 available from American National Standards Institute, 1430 Broadway, New York, New York 10018. Each separately bound report—for example, each volume in a multivolume set—shall have its unique Report Documentation Page.

1. Report Number. Each individually bound report shall carry a unique alphanumeric designation assigned by the performing organization or provided by the sponsoring organization in accordance with American National Standard ANSI Z39.23–1974, Technical Report Number (STRN). For registration of report code, contact NTIS Report Number Clearinghouse, Springfield, VA 22161. Use uppercase letters, Arabic numerals, slashes, and hyphens only, as in the following examples: FASEB/NS–75/87 and FAA/RD–75/09.

2. Leave blank.

3. Recipient's Accession Number. Reserved for use by each report recipient.

4. Title and Subtitle. Title should indicate clearly and briefly the subject coverage of the report, subordinate subtitle to the main title. When a report is prepared in more than one volume, repeat the primary title, add volume number and include subtitle for the specific volume.

5. Report Date. Each report shall carry a date indicating at least month and year. Indicate the basis on which it was selected (e.g., date of issue, date of approval, date of preparation, date published).

6. Sponsoring Agency Code. Leave blank.

7. Author(s). Give name(s) in conventional order (e.g., John R. Doe, or J. Robert Doe). List author's affiliation if it differs from the performing organization.

8. Performing Organization Report Number. Insert if performing organization wishes to assign this number.

9. Performing Organization Name and Mailing Address. Give name, street, city, state, and ZIP code. List no more than two levels of an organizational hierarchy. Display the name of the organization exactly as it should appear in Government indexes such as Government Reports Announcements & Index (GRA & I).

10. Project/Task/Work Unit Number. Use the project, task and work unit numbers under which the report was prepared.

11. Contract/Grant Number. Insert contract or grant number under which report was prepared.

12. Sponsoring Agency Name and Mailing Address. Include ZIP code. Cite main sponsors.

13. Type of Report and Period Covered. State interim, final, etc., and, if applicable, inclusive dates.

14. Performing Organization Code. Leave blank.

15. Supplementary Notes. Enter information not included elsewhere but useful, such as: Prepared in cooperation with . . . Translation of . . . Presented at conference of . . . To be published in . . . When a report is revised, include a statement whether the new report supersedes or supplements the older report.

16. Abstract. Include a brief (200 words or less) factual summary of the most significant information contained in the report. If the report contains a significant bibliography or literature survey, mention it here.

17. Document Analysis. (a). Descriptors. Select from the Thesaurus of Engineering and Scientific Terms the proper authorized terms that identify the major concept of the research and are sufficiently specific and precise to be used as index entries for cataloging.

    (b) Identifiers and Open-Ended Terms. Use identifiers for project names, code names, equipment designators, etc. Use open-ended terms written in descriptor form for those subjects for which no descriptor exists.

    (c). COSATI Field/Group. Field and Group assignments are to be taken from the 1964 COSATI Subject Category List. Since the majority of documents are multidisciplinary in nature, the primary Field/Group assignment(s) will be the specific discipline, area of human endeavor, or type of physical object. The application(s) will be cross-referenced with secondary Field/Group assignments that will follow the primary posting(s).

18. Distribution Statement. Denote public releasability, for example "Release unlimited", or limitation for reasons other than security. Cite any availability to the public, with address, order number and price, if known.

19. & 20. Security Classification. Enter U.S. Security Classification in accordance with U.S. Security Regulations (i.e., UNCLASSIFIED).

21. Number of pages. Insert the total number of pages, including introductory pages, but excluding distribution list, if any.

22. Price. Enter price in paper copy (PC) and/or microfiche (MF) if known.

UKTC-TR-01/87

# ARTIFICIAL INTELLIGENCE IN THE ALLOCATION OF MAINTENANCE RESOURCES FOR INTELLIGENCE SYSTEMS

by

Chip McConville and Paul Orgren
Unisys Corporation
Reston, Virginia

**Abstract**

Commanders of Army military intelligence units in a tactical environment are responsible for keeping their assigned equipment maintained and in operation since early replacement in a conflict may not be possible. Scheduling maintenance assets to repair and service assigned equipment is critical to success. This paper discusses the development of an expert system to assist the commander in the proper allocation of his maintenance resources. Since this expert system assists in the resolution of an allocation of resources or scheduling problem, insight into the application of artificial intelligence to this type of problem is gained.

## 1 Introduction

The allocation of resources to solve a particular problem has plagued managers since before the Pharaoh of Egypt mistakenly assigned the wrong workmen and allocated too little straw for the bricks while building the pyramids. Today, military commanders and their staffs are applying both rudimentary and sophisticated tools to deal with complex scheduling problems in an environment that is potentially more hostile than the plagues that the Pharaoh was forced to suffer through due to his scheduling mistakes. This paper discusses the development of a scheduling tool to be used by the commander and staff of a military intelligence unit in order to assist the commander in allocating his maintenance resources. In addition, this paper provides some insight into possible resolution for a problem in a job shop type environment. The specific problem addressed involves the scheduling of repairmen and repair assets to repair equipment that may or may not be at the same location on the battlefield. Currently, the scheduling is accomplished by various supervisors in a nightly conference and updated during the day by maintenance supervisors. The process is entirely manual and relies heavily on the expertise of maintenance supervisors. The approach taken to solve this problem is to build an expert system that captures the expertise of the various supervisors and applies this expertise against a corporate data base to automatically generate a schedule that can be rapidly updated as the battlefield situation changes. The first section in the paper discusses scheduling problems in order to provide the reader a general background in scheduling. The second section describes an artificial intelligence tool which is being applied to the problem in order to facilitate a solution. Next, there is a description of the actual problem domain and the specific solution that is proposed. The last two sections address future directions for artificial intelligence in attacking scheduling problems and a summary of the entire paper.

## 2 Background

### 2.1 Scheduling problems in general

To understand the solution to a specific scheduling problem, a discussion of definitions and scheduling problems in general is necessary. A scheduling problem is one in which people, time and resources are assigned to subtasks in order to complete a task. The assignments must meet certain constraints such as completion date, total cost to complete, or resource limitations. A typical scheduling problem is that of project or program scheduling, where there is a fairly good idea at the start what the activities or subtasks will be, how long each will take, what resources are required, and which subtasks are prerequisite to others. This type of scheduling is reasonably straightforward and is relatively static. The schedule may have to be adjusted because of emergencies or because subtasks take longer than expected, but this rescheduling may be a once a week or once a month procedure. It can be handled by pencil and paper if small enough, by a batch computer scheduling program if larger, or by an interactive program if quicker response is desired.

### 2.2 Job shop scheduling problems

A more difficult class of scheduling problems is the job shop problem. The job shop problem is defined by operations researchers to be the scheduling of a given number of jobs on a given number of machines when some jobs are required to be processed on several machines possibly in a required sequence[3]. The machines or resources may either be in a fixed location such as a drill press in an aircraft rebuild facility or in a mobile configuration such as repair trucks being dispatched to inspect and/or replace artillery tubes. Also a job shop scheduling problem normally has some extra complications. For example, a repair facility may repair numerous unique pieces of equipment, job orders for varying quantities and types of equipment could arrive continuously, and different jobs may require different sets of resources (human and machine) to repair. In this environment, a schedule for a job shop problem frequently requires adjustment as new jobs arrive and resources become temporarily unavailable. This dynamic situation makes scheduling a complex process. While some problems may not meet the exact definition of a job shop problem, numerous problems fall within the context of a job shop environment.

Additionally, job shop scheduling is NP-hard[2] which means that the time required to obtain an optimal solution grows exponentially with the size of the problem. What makes it tractable at all is the willingness to accept reasonable solutions (rather than the optimum solution), and the use of constraints. By providing appropriate constraining factors, such as due dates, suitability of resources for certain jobs, and intuitive restrictions, usable solutions can be reached. [1], [10]

### 2.3 Scheduling problems in the military

The military has many project scheduling type problems. These range from the program manager of a major weapons system scheduling the developmental activities of his project to a battalion operations officer scheduling the activities associated with moving a battalion by aircraft to a remote location. The military also has numerous job shop type problems. In a sense, combat itself is a job shop environment in which the commander's units can perform a number of different types of actions in order to do numerous jobs or missions but must perform operations within missions in some required sequence in order to be effective. The specific problem in this paper requires that repair assets be assigned to jobs in a rapidly changing environment. Some jobs must go through two or more sets of repairmen or repair equipment, but that is not the normal case. Therefore, this problem is not a pure job shop problem but does fall within the scope of the job shop environment. Such job shop environment problems are not easy to solve, but realistic solutions are needed by the military.

## 3. ESP—An Object-Oriented Programming Environment

### 3.1. Overview of ESP

To develop an expert system to solve a scheduling problem, an artificial intelligence shell is required. Since the development of a tailor-made shell is both time consuming and costly, a problem-solving tool called expert system planner (ESP) has been used to develop the expert system to assist the military intelligence unit commander in scheduling his maintenance resources. ESP is an object-oriented programming environment under development at the Unisys Reston Technology Center.[9] This tool provides a mechanism for defining objects, supporting message-passing between objects, and processing passed messages. The objects in ESP are called *modules*. Although ESP runs in serial Lisp computers, modules may be thought of as independent processes (for example, an intelligence analyst might have a module for an enemy unit, and a program manager might have a module for an activity). These processes communicate with one another via ports (*hooks*) through message-passing links (*connections*).

A specific module is a member of a class (*template*). The class defines the attributes or characteristics of the module, but not the specific data associated with a particular module. When a module is defined in the ESP environment it is called an *instance* of the class. The ESP structure in which the module is defined is called a *worksheet*. The worksheet is a *window* in which the module can be displayed, moved, connected, deleted, modified, and activated (these actions are called *module operations*). A collection of modules assembled together and coordinated with each other on a worksheet is the basis for problem solution in ESP.

Module operations in ESP are menu-selectable. When operations (e.g. move, delete) are selected from the Module Operations Menu, they are applied to specific modules which are displayed on the worksheet. The modules and their hooks are usually *mouse-sensitive* (i.e. when the cursor is positioned near an object, a rectangle appears and encloses the object) and can be selected for an operation by clicking a mouse button when mouse-sensitivity is indicated.

Modules and worksheets that have been previously defined are found in ESP *application libraries*. The master library consists of a menu of *books* of applications. When a book is menu-selected, the modules and worksheets in that book appear and can be subsequently selected. Worksheets can be selected from the application library for further development or just for running, and modules can be selected for instantiation into a worksheet.

### 3.2 Strengths of ESP for scheduling problems

ESP is a graphically oriented tool that has the characteristics of reusability, adaptability, and extensibility. The library of modules encourages developers to use existing modules when possible thereby providing for reusablility. New modules are also easily built on existing modules, so that the library is extended into new areas. For example, in building a job shop scheduling activity, a generic activity module can be selected from the library, modified slightly, and added to the library as a job shop activity (extensibility).

Using ESP to solve or model a problem is usually done incrementally which makes the shell quite adaptable. That is, a few modules can be selected (from the existing library of modules) or developed (and added to the module library) and linked together to form the first model of the problem. From experience with that model, new modules can be built on the original ones, and other modules can be selected or developed. By doing this step-wise refinement, the model can approach as closely as desired the problem being modelled. Thus the user is not faced with an all or nothing task, and different experts can develop different parts of the model and include their knowledge of the problem within the modules they are working on in a distributed fashion. In particular, with a scheduling problem, the knowledge engineer might first construct the schedule layout and display modules. The expert could then manipulate schedules to better understand what factors he takes into account in generating the schedules. Experience gained from that can be used in creating an automatic scheduler.

Since job shop scheduling is a problem whose complexity increases exponentially with the size of the problem, the optimal solution is not readily obtainable in the general case. However, once heuristics are used to generate a more or less acceptable solution, ESP with its visual modelling provides the tools necessary for a human to see the proposed solution and easily try out variations to the computer generated solution which incorporates the knowledge of the scheduling expert. This graphical nature of ESP makes it well suited to scheduling applications.

# 4 Example Problem—Tracking And Planning System (TAPS)

## 4.1 The domain of the problem

The solution to any scheduling problem depends upon the the entities involved in the scheduling environment and the constraints placed upon that environment. The problem for this paper involves Army military intelligence (MI) commanders in a tactical environment who are responsible for keeping their assigned equipment maintained and in operation since early replacement during combat may not be possible. These Combat Electronic Warfare and Intelligence (CEWI) commanders are assigned maintenance units of two basic types to accomplish this task. The first type is a mechanical maintenance unit which provides organizational maintenance support for such items as vehicles, power generators and air conditioners. Maintenance on these common items of equipment follows the Army's three category system of maintenance (organizational, intermediate, and depot) with the CEWI unit only being responsible for organizational level maintenance. The second type is a communications-electronic (C-E) maintenance unit which provides organizational and intermediate maintenance support to the unique electronic equipment found within the CEWI unit.[6]

To keep this equipment maintained, the CEWI commander follows the Army's forward support concept. This concept stresses equipment be repaired as far forward as possible to reduce the time required to return it to its tactical position. Normally, MI assets are deployed well forward and unit maintenance contact teams from either the mechanical or C-E unit go forward and attempt to fix the equipment on-site. When on-site repair is not practical, equipment is removed from forward combat areas to safer locations and immediate repairs made. Equipment which cannot be repaired is evacuated to the next higher maintenance unit.[6] Preventive maintenance is also performed to prevent future failures and extend the life of the equipment. This servicing of equipment is essential during any protracted conflict or training exercise. In addition, certain types of maintenance actions can be deferred until parts or repairmen become available. This deferred maintenance must also be planned and accomplished to maximize the effectiveness of combat systems.[5]

Today, the scheduling of maintenance during field operations is accomplished by a collective effort of unit leaders and maintenance supervisors during a nightly conference. This scheduling conference may take place at a fixed site or may be accomplished on the radio if conditions dictate. Since tactics require radio transmissions of short duration to prevent jamming and direction finding, significant degradation due to incomplete information occurs when only the radio is used to conduct scheduling. The end product of the scheduling conference is a paper schedule that lists the jobs to be done the next day, the repairman who will do the job and the order in which the jobs must be accomplished to prevent improper sequencing. (Improper sequencing could occur when a radio repairman arrives to repair a radio at a remote site and finds out that the vehicle has been evacuated to another location for repair by a vehicle mechanic or that the vehicle does not have any power to test the radio since the vehicle mechanic has not repaired the vehicle yet.) The schedule does not attempt to take into account the availability of other repair assets such as parts or test equipment. The current system relies on the memory of the the mechanical and C-E maintenance supervisors to incorporate such details into the decision making process. Once the schedule is completed, the responsibility for keeping the schedule updated as additional equipment requires repair or repair assets become unavailable belongs to the maintenance supervisors and is coordinated through the unit maintenance officer.

## 4.2 The constraints and needs of the problem

Some major constraints must be taken into consideration by the individuals conducting maintenance scheduling within the CEWI unit. These constraints include the availability of: repairmen with the correct skills, repair parts, tools or test equipment, and adequate time and transportation to perform all required repairs and services. Additionally, the scope of the scheduling problem for a CEWI company will include the following maintenance resources: 33 repairmen, 66 major systems with numerous associated subsystems, representative repair equipment and parts, 26 movable repair locations and six (6) movable evacuation sites.[7] In order to assist the CEWI commander and his staff in properly allocating these scarce resources, an expert system that will act as a decision aid is needed. This expert system will consolidate the information available from the corporate data base and the heuristics involved in the scheduling process[4] and automatically generate a schedule that meets the constraints of the problem or alerts the supervisor that constraints must be relaxed to generate a feasible schedule. The expert system must be readily adaptable to changes dictated by both changes in the battlefield and changes generated by supervisors. In order to accomplish this, the following objectives and resulting goals must be met:

1. Provide a decision aid for the CEWI commander and his staff which will:
   (a) Schedule repairmen to jobs in a dynamic environment both on a daily and case-by-case basis thereby reducing the overall down time for systems.
   (b) Maintain information on the status of all repair assets and assist the maintenance supervisors in their appropriate usage thereby reducing excess parts and tool losses within the unit.
   (c) Consolidate maintenance tasks when appropriate thereby reducing the overall repair time for all jobs (deadline jobs, services and deferred maintenance).
2. Provide a graphical tracking system to maintain control of contact teams deployed to forward locations thereby keeping these teams in friendly areas and incorporating a means for the supervisors to improve their routing between jobs when necessary.

This expert system must also fit the current tactical environment and be able to be used by both the commander and his maintenance supervisors.

### 4.3 Proposed solution

The approach to solving this problem involves the development of modules and submodules to properly represent the scheduling environment using ESP. These modules are tied together in the supervisor's interface using module operations to properly connect, define, display and manipulate the modules. Each module developed directly relates to an objective the expert system must meet. The overall objective is to develop a system that automatically generates a schedule with the option of further modification to the schedule and knowledge base which includes both the corporate data base and a set of heuristic rules.

SUPERVISOR'S INTERFACE: The purpose of the supervisor's interface is to provide a means through which the maintenance supervisor can interface with the system to generate a schedule. Through this interface, the supervisor manipulates modules to update the situation and then generates other schedules or queries the corporate data base for information he needs to better manage his environment. He inputs this knowledge via specially designed pop-up menus, command windows and system prompts with default values provided. Information is provided to the supervisor through both tabular and graphical displays. The central graphical display shows the tactical disposition of units, repair assets and systems to be repaired. Graphical displays are also used to display schedule information as well as percentage of asset usage. These graphical displays are in the form of worksheets with submodules that the supervisor has defined. He manipulates these submodules on the worksheet to portray the current situation and may project future schedules under varying situations when necessary. Additional information can be obtained through a mouse and menu selection system specifically designed to quickly provide the supervisor with information that he desires. This information is normally of a status type nature such as the location or activity of a given repairman. Each worksheet can have different levels of detail displayed through a layering technique. The supervisor merely defines which modules or submodules are to be placed on a given layer and then uses a mouse and menu to select the layers he desires to be displayed at any given time. The worksheet with layers technique allows the supervisor's interface to be used to access all modules within the expert system. Figure 1 shows a supervisor's display with multiple layers which include both the data base and map modules.

JOB SCHEDULER MODULE: This module is designed to meet the objective of scheduling repairmen and repair assets to jobs in a dynamic environment both on a daily and case-by-case basis. The job scheduler module consists of activities that are placed on a time line on one of the maintenance supervisor's layers of his worksheet. The activities are jobs that require scheduling and the time line runs for a 24 hour period. Constraint propagation is used to help generate a viable schedule with the constraints embedded within the activities. Additional scheduling rules can be found within the rule base module. This module controls the automatic generation of a daily schedule. Since the supervisor may have some intuitive constraints or information that is not included in the system, the ability to modify the schedule and generate a new schedule on a case-by-case basis is provided. To revise the schedule, the supervisor can move the jobs to a new position on the screen using the mouse. This movement can reflect either assigning the job to another repair asset or placing it in a new sequence. Once he is satisfied with the new schedule, the supervisor can execute a run function which checks to see if the schedule meets all constraints and also generates several measures of effectiveness for comparison with the old schedule. These measures are total completion time, percentage of jobs scheduled, number of deferred maintenance jobs scheduled, and wait time for high priority jobs. Figure 2 is an example of the job scheduler display, showing the activities to be scheduled and the measures of effectiveness.

DATA BASE MODULE: This module is designed to meet the objective of maintaining information on the status of all repair assets and assisting the maintenance supervisors in their usage of these assets. This module also provides access to the corporate data base for the remainder of the modules. Initially, the input of the data will be manual through the supervisor's interface, but eventually automatic interfacing with automated systems containing required data is envisioned. The data base module both creates submodules and stores data concerning these submodules for use by the other modules or for the maintenance supervisor to query. For example, the job scheduler module accesses the database module which contains a submodule or record type called repair parts. The various parts themselves are instances of this record type. Therefore, the data base module is able to keep track of all the individual parts that are entered into the system. Initially, these parts will be entered manually through a mouse and menu technique by the maintenance supervisor. This supervisor can then query the data base for the status of any item by either clicking the mouse on the icon representing the item on the worksheet since these submodules know how to display themselves, or he can query the database directly by accessing the specific record type of interest using a mouse and menu technique. The data base module must also permit searching records, establishing sets of data, and defining relationships between record types.

RULE BASE MODULE: This module is designed to meet the objective of consolidating maintenance tasks when appropriate and to incorporate the surface knowledge[4] provided by the maintenance supervisor expert in the form of heuristics. These heuristics must be turned into a set of rules that are coded in Lisp and included in a module that communicates with the other modules in the system through message passing to obtain both information such as the location of a given submodule or when to trigger a given rule. The set of rules must incorporate such general scheduling rules as schedule all deadline jobs prior to any other type, or if a specific vehicle requires a service check to see if any deferred maintenance is required and when true perform this maintenance when several other resource constraints are met.

MAP MODULE: This module is designed to meet the objective of providing a graphical tracking system to maintain control of contact teams deployed to forward areas as well as provide a locational reference system for other modules and submodules to access. The map module provides a grid reference system on any required worksheet. Since other modules or submodules can have values within their instances for a location, this location value can be used by the map module to locate the proper position on the screen. Therefore, on any worksheet that the maintenance supervisor views, he is able to properly position an asset or find out the current grid location of an asset by executing a run operation for the map module and placing the cursor of the mouse over the item of interest. This will cause the actual grid coordinate to be displayed on the screen. To aid in keeping maintenance contact teams in friendly areas, a submodule for the frontline positions can be automatically created using graphic item modules. These graphic items are available through ESP to create tailored graphic modules on the worksheet. A polyline tracing the frontline position is created as a module so through a message passing technique between modules warnings are issued if contact teams move or are assigned to move past frontline positions. The map module also contains the capabilities to rescale the map, scroll through a large area, or redefine grid line spacing.

### 4.4 Current status of solution

Currently, a preliminary system using a portion of an actual data base is under development on a Lisp work station. Incremental development is being used to develop each module and connect them as appropriate. The supervisor's interface is functioning to create, display and manipulate the basic modules. Additionally, the layering technique for this interface is totally functional. The job scheduler module has been developed to the point that manual manipulation of a schedule is possible along a time line with appropriate measures of effectiveness automatically generated. Interfacing this module with the other modules is a high priority. The data base module is totally functional, and a basic set of data has been entered. The objects or submodules within the data base are being displayed on the supervisor's interface and can be manipulated or queried as desired. The rule base module is still in the knowledge acquisition stage with various rules being developed so that the scheduling module can move from the current manual mode to an automatic mode as quickly as feasible. The map module is totally functional to include incorporation of a graphic editor for drawing overlays to reflect the current tactical situation. Coding to interface this module with the data base module to continuously track the location of contact teams and provide appropriate warnings is in progress. Overall, the preliminary system is developing as expected with no major problem areas encountered thus far.

## 5 Future Directions

### 5.1 Future enhancements of ESP for scheduling problems

Although ESP is being used to help solve a complex scheduling problem in its current configuration, a few enhancements to the shell would be useful. One of the most important features needed in ESP for scheduling problems is an interface to existing data bases. Much of the information needed to do scheduling, such as job orders received, repairmen available, and inoperable equipment, are already recorded in data bases. Currently, solutions require that the information be manually entered into ESP. It is obvious that for a system to be in daily use, a direct connection between ESP and existing data is vital.

The best implementation of the automatic scheduling portion of an expert system is probably a rule-based inference system. At the present time, the two choices of inference engine that a developer using ESP has are to link to OPS-5[4], Prolog, or to write his own inference engine as done in TAPS. It would be useful to have a wider range of inference engines built into ESP and integrated into the system so that modules have easy and straightforward access to them.

Another improvement of more general utility would be an automatic module generator. Currently, although modules can be used, connected, and manipulated by people with no programming skills, to build a module in the first place requires someone who has some proficiency in Lisp. A module generator that could turn a higher level description into a module would simplify the construction of scheduling solutions and allow domain experts with no Lisp experience to be quite effective in developing scheduling expert systems.

### 5.2 Future of artificial intelligence in military scheduling

TAPS reflects only a very small portion of the need for expert systems in military scheduling. Today, the major problem facing the military commander is the ability to apply combat power at the precise time and place desired. In order to do this, the airland battle doctrine was developed. This doctrine calls for seizing the initiative and exercising it aggressively to accomplish the mission. Success on the future land battlefield will depend on the armed force's ability to fight in accordance with four basic tenets: initiative, agility, depth and synchronization.[8] The last of these, synchronization, holds the most promise to be aided by the judicious application of computer technology. Since "synchronization is the arrangement of battlefield activities in time, space and purpose to produce maximum relative combat power at the decisive point"[8], expert systems to assist the commander in scheduling these battlefield activities will have a high payoff. Such systems must allow the commander and his staff to manipulate battlefield objects in both space and time so that the purpose of each object can be clearly defined and time and space constraints clearly understood. Artificial intelligence can play the key role in providing such scheduling expert systems. To do this, artificial intelligence must continue to incorporate more and more of the commander's environment in a realistic fashion within its methods and provide for usable natural language interfaces.

## 6 Conclusions

The allocation of maintenance resources to maintain military intelligence systems is a complex process. The commander and staff of the unit responsible for such maintenance is faced with a difficult scheduling environment. This paper has described his problem and provided a possible resolution to his problem by using an expert system called TAPS which was developed using an artificial intelligence shell called ESP. ESP is an artificial intelligence tool that is graphically oriented and based upon a system of modules that portray the real world scheduling environment as realistically as possible. Enhancements and further work are necessary to improve and develop both ESP and TAPS; however, the solution to the problem is within reach today and better solutions for more complex problems will exist within the discipline of artificial intelligence tomorrow.

## Acknowledgements

The authors gratefully acknowledge the efforts of Fred King and Eric Hushebeck for their work in developing ESP and their assistance in writing this paper and on the TAPS project. We would also like to acknowledge both Unisys and the U.S. Army's Artificial Intelligence Center for their mutual support of a training with industry program that made this project possible.

## Disclaimer

Material in this paper should be considered the authors' personal views and not the official views or policy of the U.S. Army or Unisys.

## References

1. Fox, M.S., B.P. Allen, and G.A. Strohm, "Job Shop Scheduling: An Investigation in Constraint-Directed Reasoning," *Proceedings of the Second National Conference on Artificial Intelligence*, 1982, pp. 155-158.

2. Garey, M.R., and D.S. Johnson, *Computers and Intractability*, Freeman, 1979.

3. Giffin, Walter C., *Introduction to Operations Engineering*, Irwin, 1971.

4. Harmon, Paul, and David King, *Expert Systems*, Wiley, 1985.

5. HQ, Department of the Army, **FM 29-2**, *Organizational Maintenance Operations*, 1984.

6. HQ, Department of the Army, **FM 34-1**, *Intelligence and Electronic Warfare Operations*, 1984.

7. HQ, Department of the Army, **FM 34-30**, *Military Intelligence Company (Combat Electronic Warfare and Intelligence)*, 1983.

8. HQ, Department of the Army, **FM 100-5**, *Operations*, 1986.

9. King, J. Fred and Eric Hushebeck, "ESP—A Knowledge-Aided Design Tool," *Proceedings of the 1986 IEEE International Conference on System, Man, and Cybernetics*, Atlanta, Ga., Vol 2, October 1986, pp. 1319-1324.

10. Smith, S.F., Mark S. Fox, and Peng Si Ow, "Constructing and Maintaining Detailed Production Plans: Investigations into the Development of Knowledge-Based Factory Scheduling Systems," *AI Magazine*, Fall 1986, pp. 45-61.

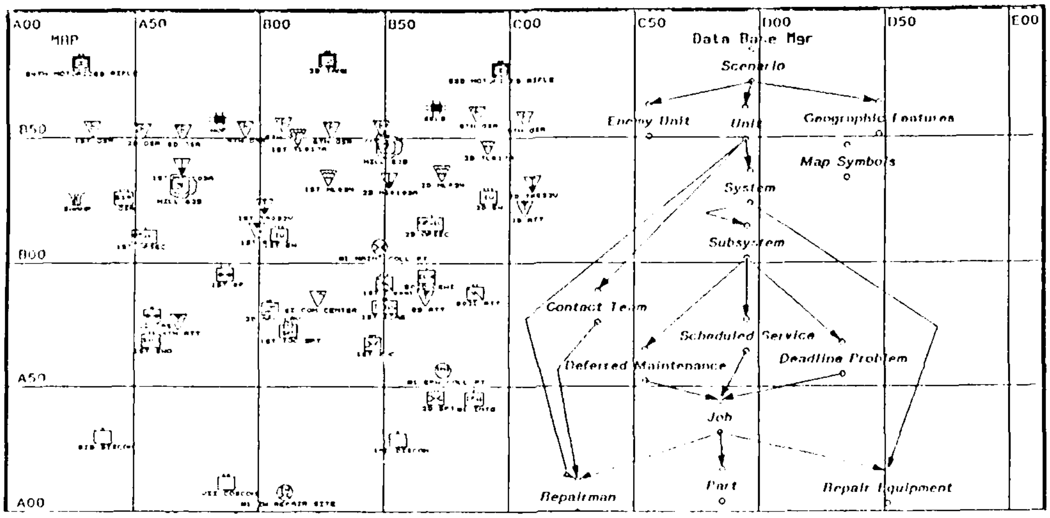


**Figure 1.** Multiple layers displayed in a user familiar environment by the supervisor's interface.

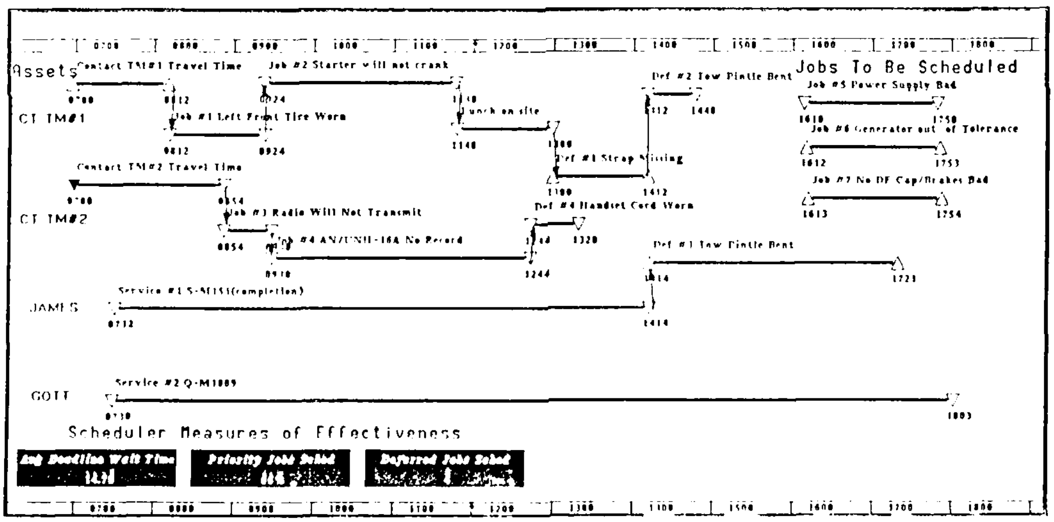


**Figure 2.** The scheduling layer, showing a feasible schedule with appropriate measures of effectiveness.